Beyond Accuracy: Comparative Analysis of Neural Network Architectures, Explainability, and Robustness in Yoga Pose Classification

Tomer Ben Aroush 209157692 , Idan Avisar, 313321903
Deep Learning 046211

1. Introduction

In real-world domains such as sports analytics, healthcare, and surveillance, models must be both accurate and trustworthy, making decisions based on meaningful features and remaining robust to noise or distortions.

1.1 Project Goal: This work aims to systematically compare multiple neural network architectures for yoga pose classification, focusing not only on accuracy but also on explainability and robustness. We seek to understand how models make decisions, where they focus, and how well they withstand data degradation from Gaussian noise.

1.2 Motivation: Accuracy alone can be misleading, as a model may rely on irrelevant features or fail under noisy conditions. Combining explainability with robustness testing addresses both interpretability and reliability, enabling better architecture selection for practical applications. This project, titled *"Beyond Accuracy: Comparative Analysis of Neural Network Architectures, Explainability, and Robustness in Yoga Pose Classification"*, aims to systematically evaluate and compare multiple well-known neural network architectures for the task of yoga pose classification. **Our primary goal** is to go beyond the standard accuracy metric and explore:

1. **Comparative analysis**: of accuracy, hyperparameters, and other metrics.
2. **Explainability**: using Grad-CAM and Grad-CAM++ to visualize model focus. In Grad-CAM visualizations, the heatmap shows the areas the model focused on when making a prediction.
   Regions in **red** indicate high importance or strong activation, meaning the model relied heavily on these areas, while regions in **blue** indicate low importance or minimal contribution to the prediction.
3. **Robustness to Gaussian noise**: testing under varying levels of Gaussian noise.

The dataset contains five yoga poses split into training, validation, and test sets, with augmentations such as random cropping and flipping applied for generalization. This work provides a holistic view of pose classification performance, emphasizing both interpretability and resilience

1.3 Previous Work

Human pose recognition has been extensively studied, with CNNs like VGG16 and ResNet achieving strong results, while lightweight CNNs are used for real-time efficiency. GNNs have recently been applied to pose tasks by modeling structural relationships between keypoints. Grad-CAM and Grad-CAM++ are standard explainability tools, and Gaussian noise testing is common in robustness studies. This work combines these approaches into a unified evaluation framework for yoga pose classification.

2. Method

This section outlines the methodology for training and comparing neural network architectures for yoga pose classification, including dataset preparation, preprocessing, and augmentation.

2.1 Dataset

We used a curated dataset of five yoga poses (*downdog*, *goddess*, *plank*, *tree*, *warrior2*), organized into training, validation, and test folders in PyTorch ImageFolder format. The splits were:

- **Training:** model optimization and feature learning
- **Validation:** hyperparameter tuning and early stopping
- **Test:** final evaluation

2.2 Preprocessing and Augmentation

To improve generalization and reduce overfitting, the training set was augmented with:

- RandomResizedCrop, RandomHorizontalFlip, RandomRotation
- ColorJitter for brightness/contrast variatioNormalize with dataset mean and standard deviation

Validation and test sets were resized, center cropped, and normalized for consistent evaluation. All data was loaded using DataLoader with a batch size of 32.

2.3 Model Architectures and Algorithm

The core of our methodology is a **comparative analysis** of diverse deep learning architectures to determine their effectiveness for yoga pose classification. We explored three distinct families of models: traditional Convolutional Neural Networks (CNNs), modern Vision Transformers (ViTs), and specialized Graph Neural Networks (GNNs). The eight CNN architectures analyzed were:

Convolutional Neural Networks (CNNs):
For established CNN architectures, we used transfer learning, replacing the final classifier to fit the five yoga poses.

- **AlexNet:** Early deep learning model with sequential convolution and fully connected layers.
- **VGG16:** Deep, simple design with stacked 3×3 convolutions.
- **GoogLeNet:** Uses Inception modules to capture multi-scale features efficiently.
- **ResNet18:** Employs residual connections for training deeper networks.
- **CustomCNN:** Lightweight CNN built from scratch as a baseline.

Vision Transformers (ViTs):
ViTs treat images as sequences of fixed-size patches processed by a Transformer encoder, enabling global context modeling.

- **ViT-Base:** Splits images into patches, flattens them, and uses self-attention to capture global relationships.
- **DINOv2:** Self-supervised ViT pre-trained on large unlabeled datasets, providing strong, transferable visual features.

Graph Neural Networks (GNNs):
GNNs analyze the body's structural representation rather than raw images.

- **PoseGNN:** Uses a pose estimator to extract 2D keypoints (joints) and connect them into a skeletal graph. The network classifies poses by learning relational patterns between body parts, focusing on alignment and posture.

2.4 Training and Evaluation

All models were trained using **fine-tuning** to enable a direct **performance comparison** between architectures, while CNNs were also **trained from scratch** primarily to support **explainability** analysis and evaluate **robustness to noise**, assessing the impact of pretraining.

Shared hyperparameters included:

- **Loss Function:** nn.CrossEntropyLoss, combining LogSoftmax and Negative Log-Likelihood for stable multi-class classification.

$$\mathcal{L} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_{ic}\log(\hat{y}_{ic})$$

- **Optimizer:** Adam for most models; PoseGNN used AdamW for improved results.
- **Epochs:** 50 for all models.

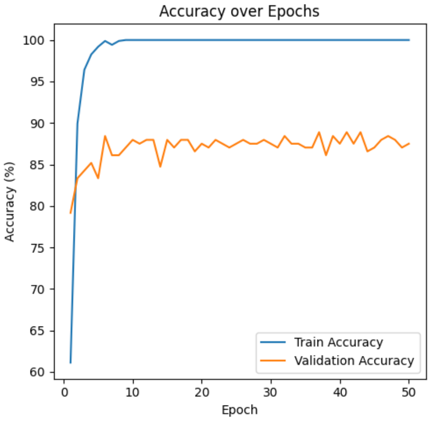
During each epoch, the models processed mini-batches, computed loss, performed backpropagation, and updated weights.

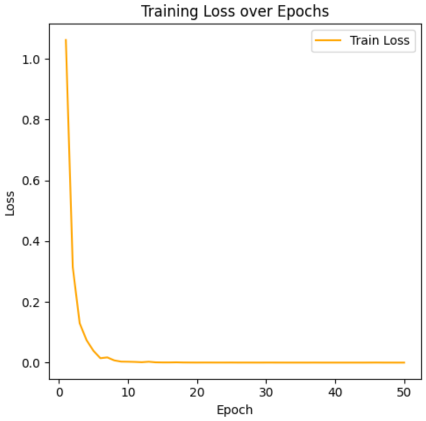
Explainability: Grad-CAM and Grad-CAM++ were used to visualize the most influential image regions for predictions, enabling verification that models focused on relevant anatomical features.

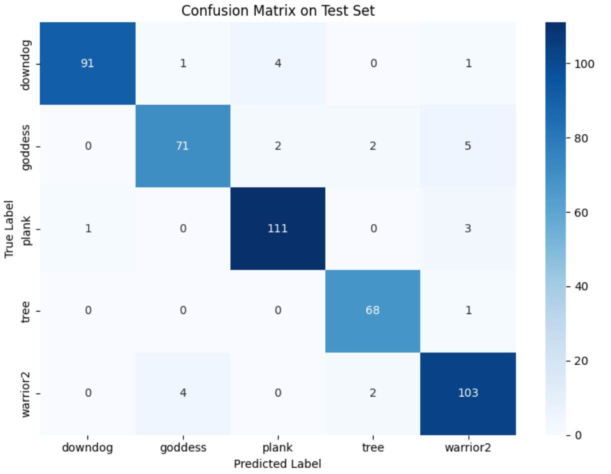
3. Experiments and Results

This section presents the performance evaluation of eight neural network architectures, validating transfer learning effectiveness and comparing them using standard metrics.

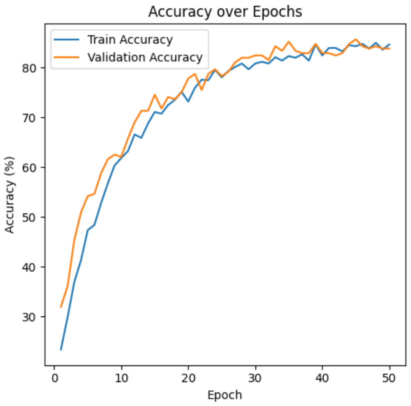
1. **VGG16**: Test Accuracy: 94.47%

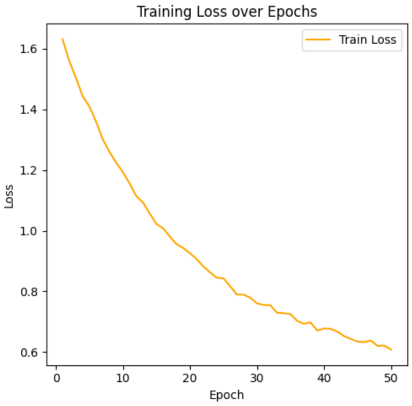


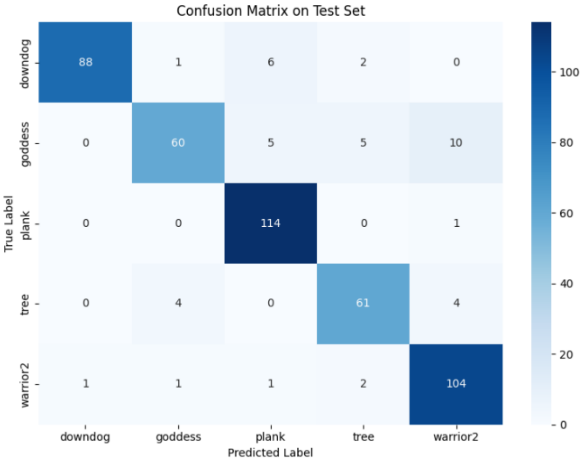




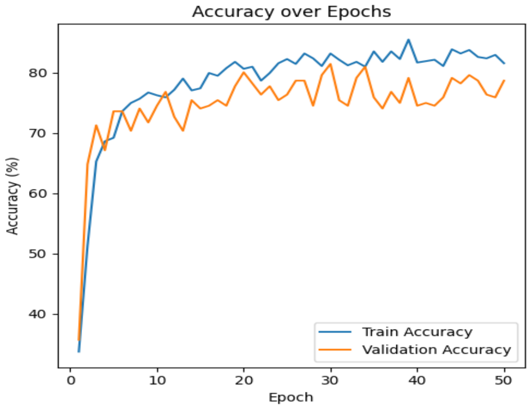
2. **ResNet18**: Test Accuracy: 90.85%

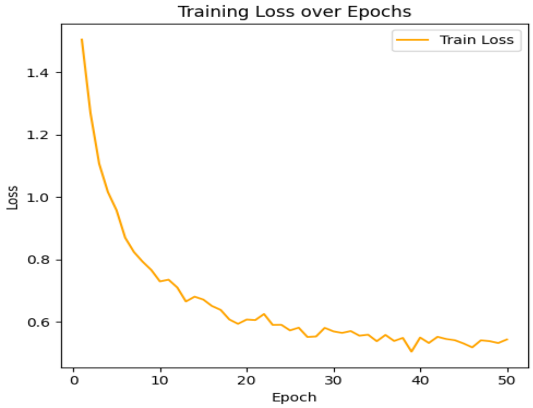




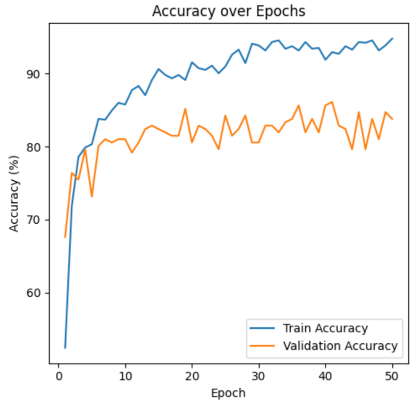


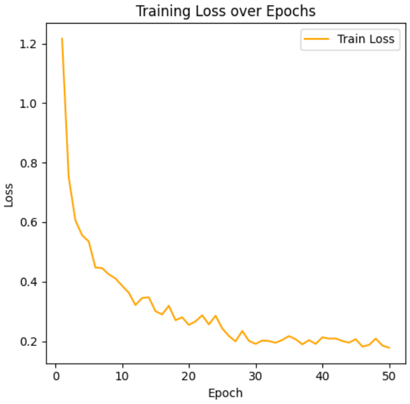
3. **GoogLeNet**: Test Accuracy: 91.28%

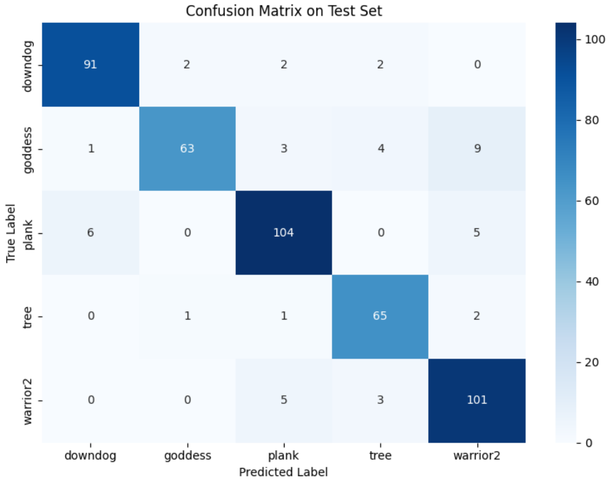




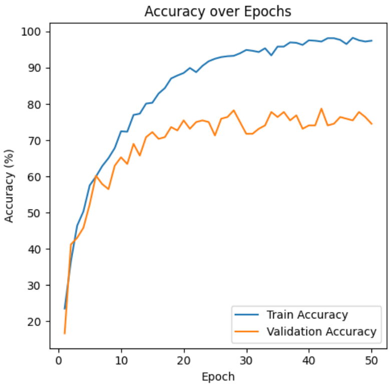
4. **AlexNet:** Test Accuracy: 90.21%

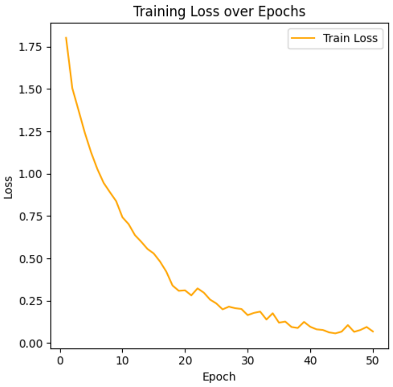


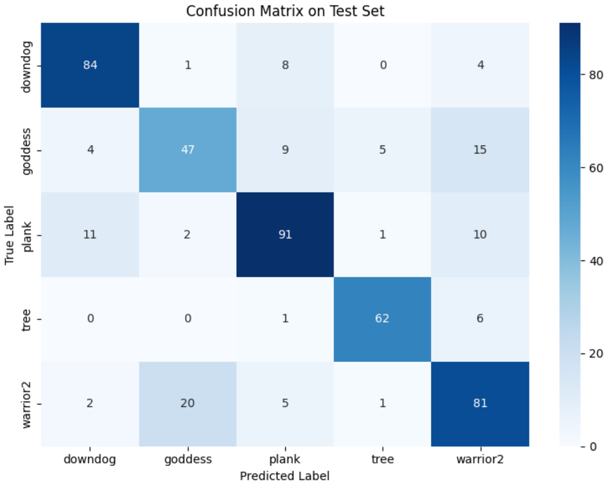




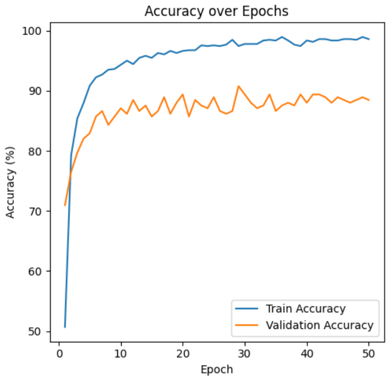
5. **Custom CNN:** Test Accuracy: 77.66%

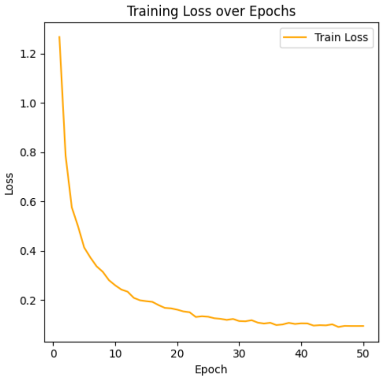


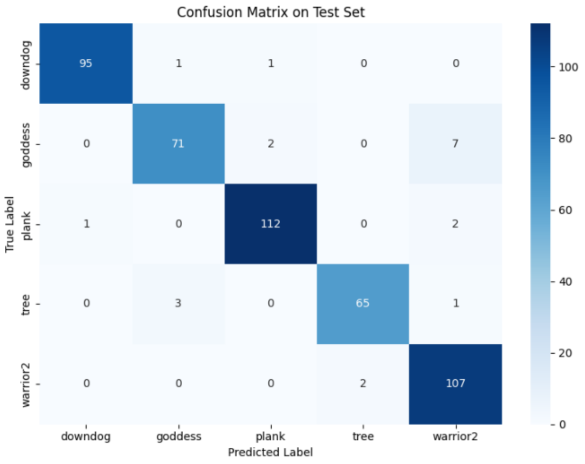




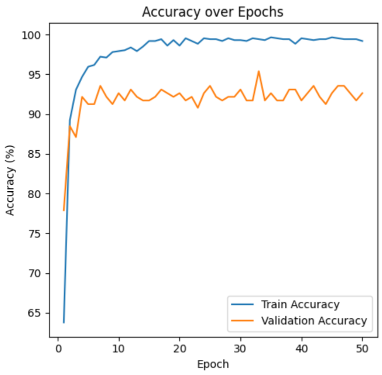
6. **ViT Base**: Test Accuracy: 95.74%

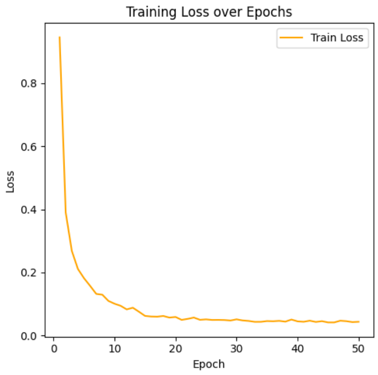


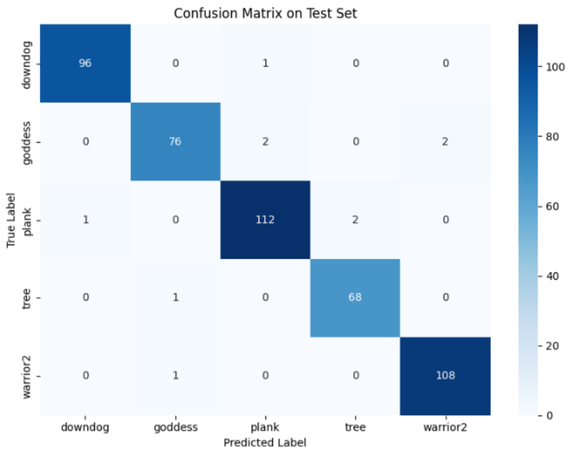




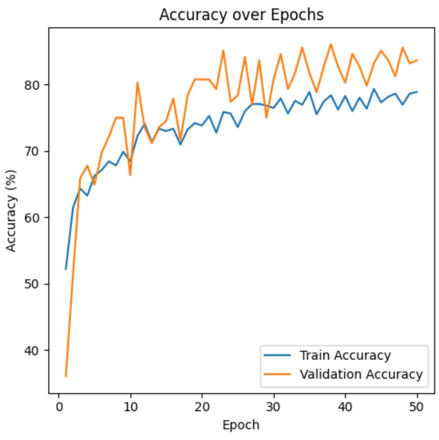
7. **DinoV2:** Test Accuracy: 97.87%

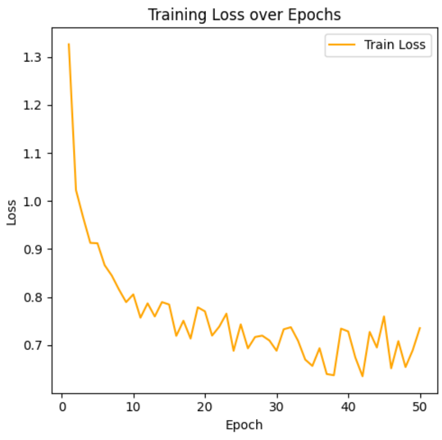






8. **GNN:** Test Accuracy: 89.03%





3.1 Performance Comparison

Architucture	Model		Test Accuracy	Precision(avg)	Recall(avg)	F1-Score(Avg)
CNN	AlexNet	Finetune	90.21%	0.9058	0.8997	0.9007
		Full train	95.74%	0.9600	0.9595	0.9597
	VGG16	Finetune	94.47%	0.9456	0.9443	0.9445
		Full train	98.51%	0.9849	0.9855	0.9851
	GoogLeNet	Finetune	91.28%			
		Full train	97.87%	0.9789	0.9752	0.9768
	ResNet18	Finetune	90.85%	0.9096	0.8973	0.9008
		Full train	98.09%	0.9806	0.9820	0.9812
	CostumCNN		77.66%	0.7796	0.7773	0.7777
ViT	ViT Base		95.74%	0.9590	0.9529	0.9555
	Dino V2		97.87%	0.9782	0.9780	0.9781
GNN	PoseGNN		89.03%			

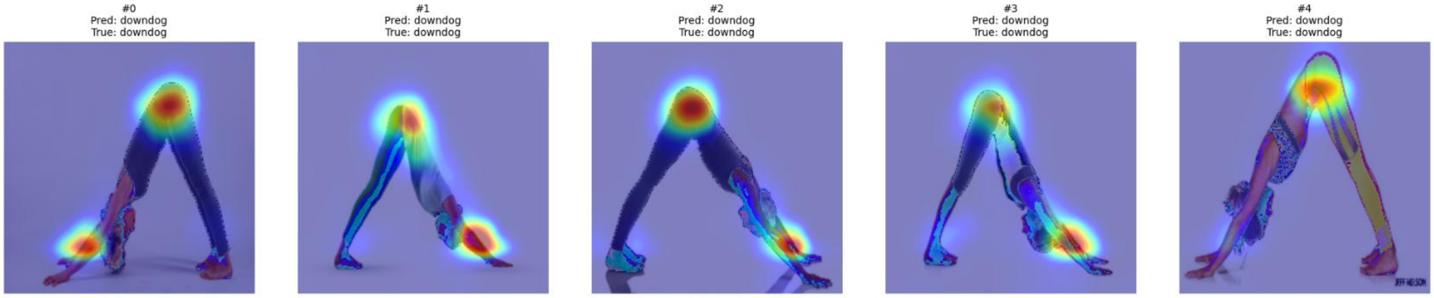
Deeper architectures delivered the best results, with **VGG16 (Full Train), ResNet18** and **DINOv2** leading their categories. VGG16's strong performance is reasonable given the dataset's limited complexity, its dense convolutional design well-suited for pose recognition, and the benefits of full training. Transformers performed well with less training, while simpler models like CustomCNN lagged behind, highlighting the importance of depth and pretraining.

Two confusion matrices are missing because the corresponding saved models did not store their weights correctly, making reproduction impossible without retraining, which was not performed.
Learning curves for CNN and ViT models showed effective training without major overfitting, and confusion matrices confirmed high accuracy with minimal misclassifications.

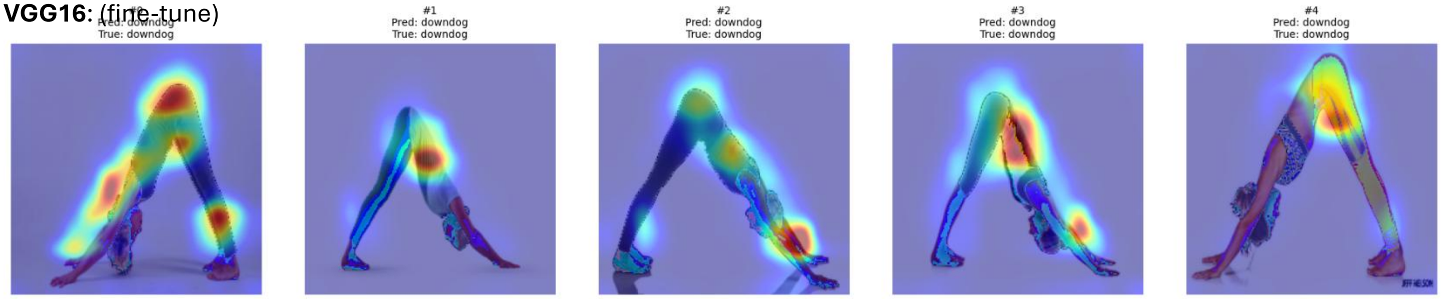
3.2 Explainability:

Here five poses from the test set, run Grad-CAM on all networks, and examine the results:

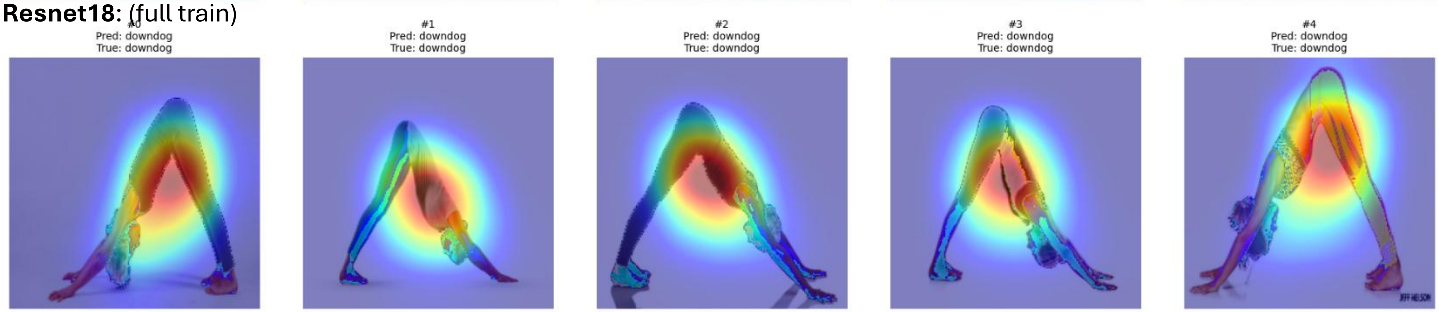
VGG16: (full train)



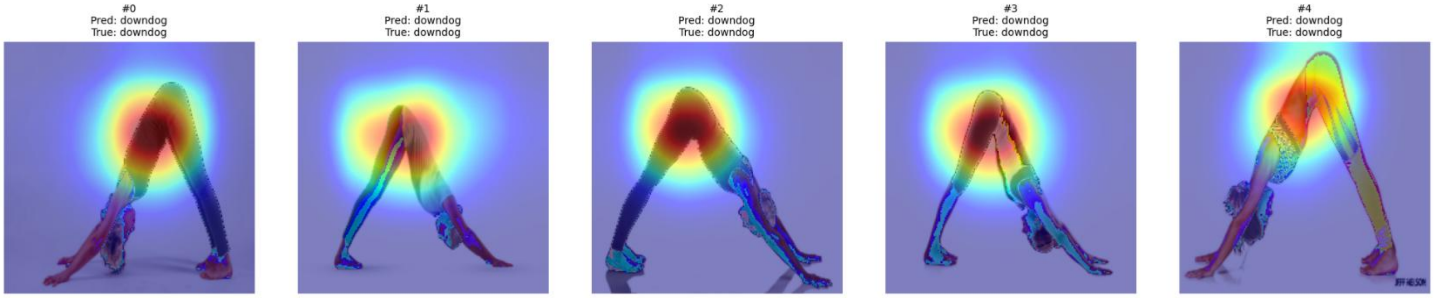
VGG16: (fine-tune)



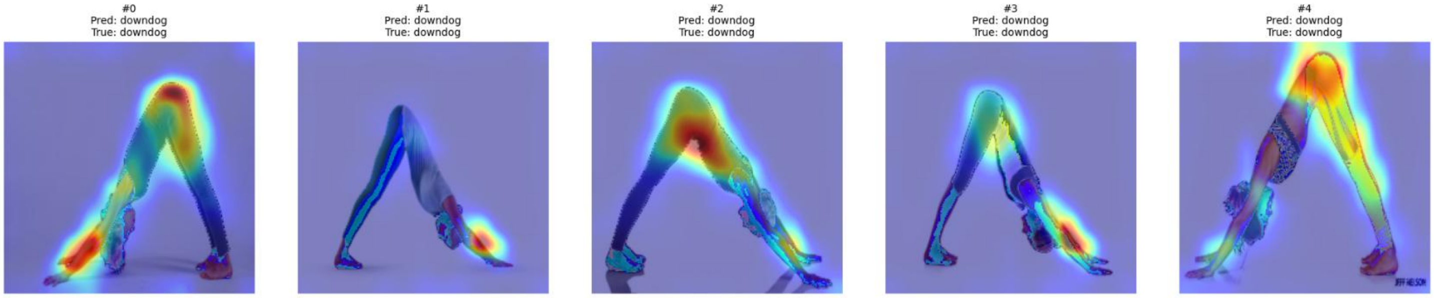
Resnet18: (full train)



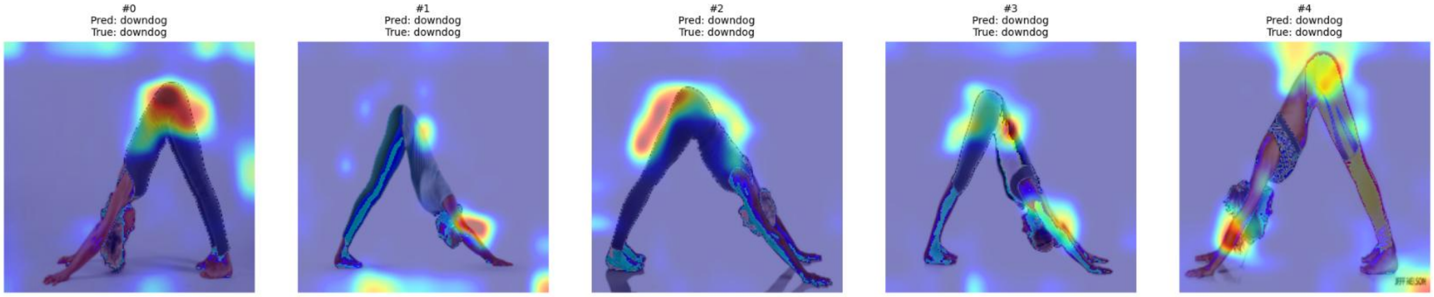
GoogLeNet: (full train)



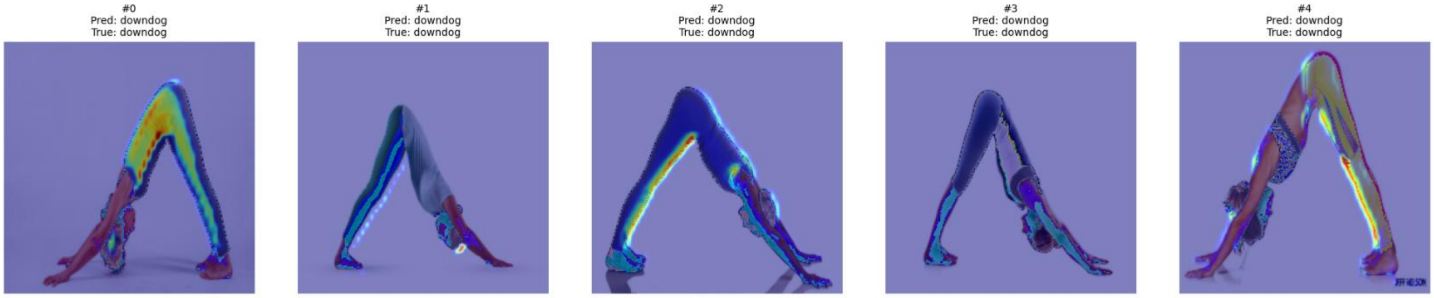
AlexNet: (full train)



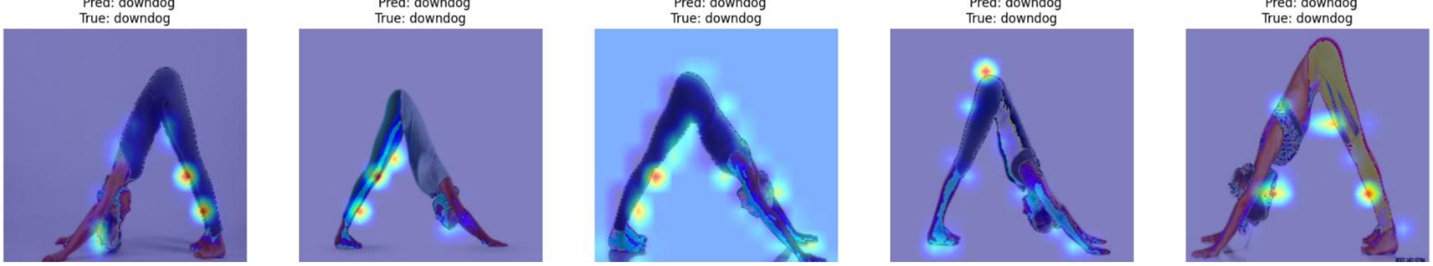
AlexNet: (fine-tune)



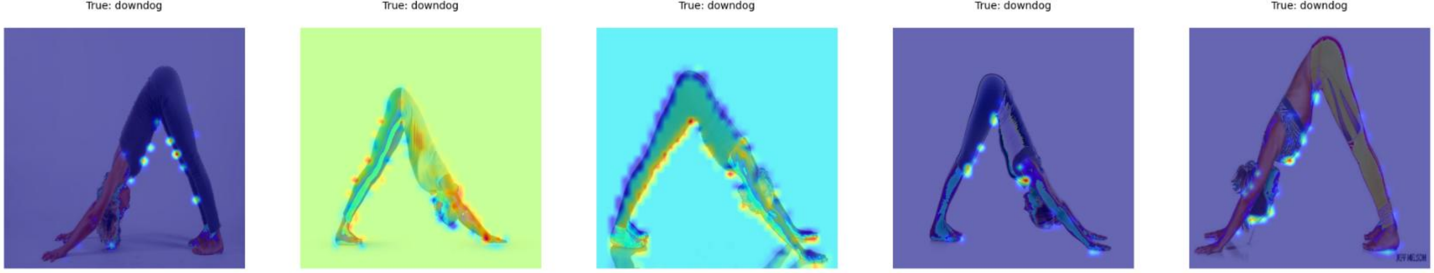
Custom CNN:



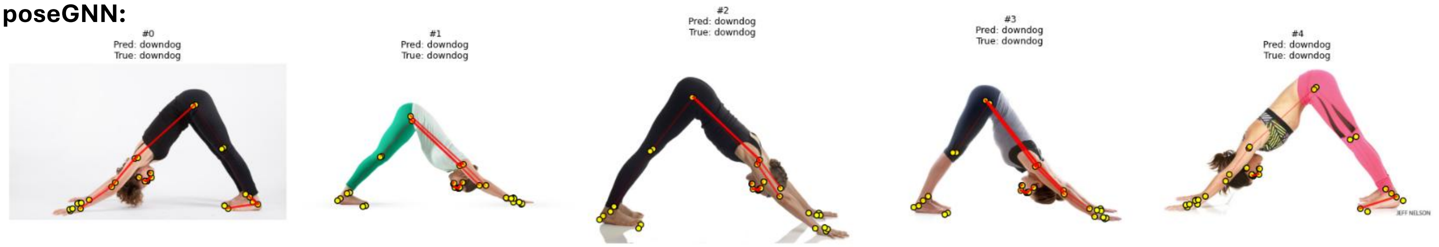
Basic ViT:



DinoV2:

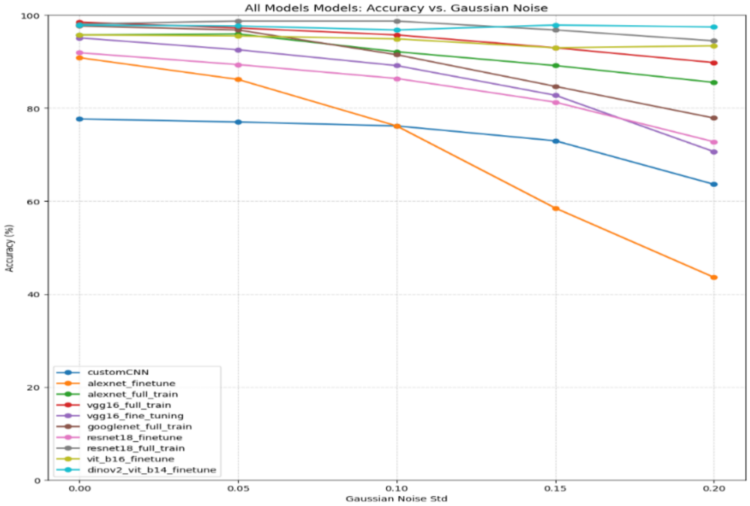


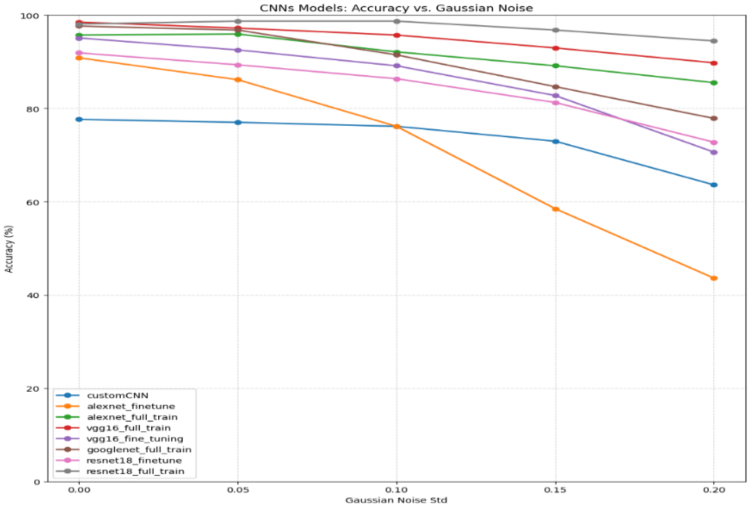
poseGNN:

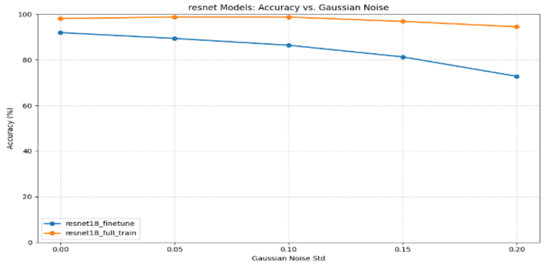


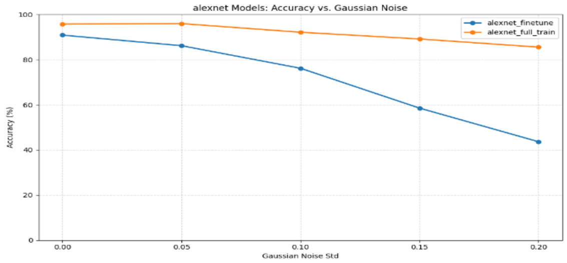
Grad-CAM for CNNs and ViTs revealed focus on relevant anatomical features (e.g., arms, hips and legs in *downdog*). PoseGNN explainability identified key joints and limbs crucial for each pose (e.g., head, hip, and foot in *downdog*), offering a more structural, human-like interpretation.

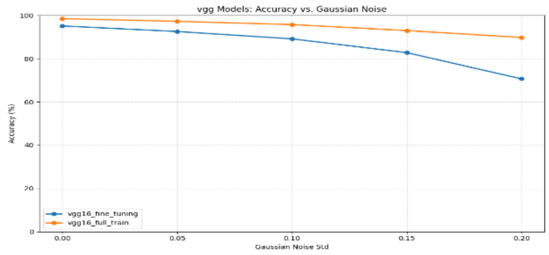
3.3 Noise Robustness

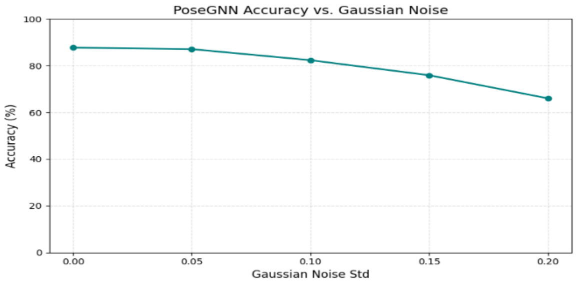












Models trained from scratch generally showed better robustness to Gaussian noise compared to fine-tuned versions, particularly among CNN architectures.
ViT models maintained competitive robustness, while PoseGNN was more sensitive to noise due to its reliance on accurate keypoint detection.

4. Conclusions

4.1 Performance Comparison

Our experiments show that deeper, well-established architectures significantly outperformed simpler ones. **VGG16 (Full Train)** achieved the highest accuracy (98.51%), which is reasonable given the dataset's characteristics and its architecture's suitability for pose recognition, followed closely by **ResNet18** at 98.09% and **DINOv2** at 97.87%.

- **Full Training vs. Fine-Tuning:** For CNN-based models (AlexNet, VGG16, GoogLeNet, ResNet18), full training from scratch consistently delivered better results than fine-tuning, in some cases improving accuracy by over 5%. This suggests that learning all parameters from scratch allows CNNs to better adapt to the specific characteristics of the yoga pose dataset, rather than relying on features learned from generic datasets like ImageNet.

- **ViT Models:** ViT-Base and DINOv2 achieved competitive results without full retraining. Their ability to model long-range dependencies between image regions proved effective for pose classification. DINOv2's self-supervised pretraining on massive unlabeled datasets contributed to its strong generalization.

- **GNN Model:** PoseGNN achieved 89.03% accuracy, outperforming the lightweight CustomCNN baseline but falling short compared to the top CNN and ViT models. This highlights both the promise and the current limitations of graph-based approaches in this domain.

- **Baseline:** The CustomCNN reached only 77.66% accuracy, reaffirming that deeper, more sophisticated architectures are necessary for high performance in complex pose classification tasks.

4.2 Explainability

Understanding why a model makes its predictions is essential for trust and model validation. We use both fully trained and fine-tuned models to observe how the accuracy, which is typically higher in the fully trained models, influences the model's attention regions as revealed by Grad-CAM.

- **CNNs:** Grad-CAM analysis for the *downdog* pose revealed clear differences in focus patterns:
  - **VGG16 (Full Train):** Strong, continuous focus on the hips, with secondary focus on arms.
  - **VGG16 (Fine-Tune):** Emphasis on back, especially the hips and lower back, with less focus as in the full train version.
  - **ResNet18 (Full Train):** Balanced focus on both upper and lower body, looking on the opposite V which defines the downdog pose.
  - **GoogLeNet (Full Train):** Centralized focus on the hip and back area and the opposite V, with weaker focus on extremities.
  - **AlexNet (Full Train):** Main focus on the hands, shoulders and back, with partial focus to the legs.
  - **AlexNet (Fine-Tune):** More scattered than the full train, Narrower focus around hips and upper legs, occasionally missing arm positioning.
  - **CustomCNN:** Dispersed and inconsistent focus, often including background regions - correlating with its lower classification accuracy.

- **ViTs:** Both ViT-Base and DINOv2 covered the entire body in their focus maps, with DINOv2 producing sharper focus on limb joints and symmetrical posture, suggesting stronger spatial awareness.

- **PoseGNN:** Instead of pixel-level heatmaps, PoseGNN identified high-importance joints such as wrists, shoulders, hips, and ankles - the critical points that define *downdog*. Its structural approach emphasizes skeletal alignment rather than texture or background.

These findings confirm that the best-performing CNNs and ViTs focus on key anatomical regions, while PoseGNN interprets the pose through joint relationships, offering complementary explainability perspectives.

4.3 Noise Robustness

We evaluated all models under varying levels of additive white Gaussian noise ($\sigma = 0.00$ to 0.20) applied to the test set.

- **Impact of Noise:** Accuracy decreased for all models as noise levels increased, confirming the importance of robustness evaluation in real-world scenarios where inputs may be degraded.

- **CNNs:** Full training from scratch generally provided more resilience to noise compared to fine-tuning. This suggests that fully trained CNNs develop more robust feature representations tailored to the target data.

- **ViTs:** Transformer-based models maintained strong robustness, likely due to their ability to model global relationships across the image, which may help them ignore localized noise patterns.

- **PoseGNN:** This model showed higher sensitivity to noise. Since it relies on accurate keypoint detection, noise in the input images can disrupt the pose estimation stage, leading to a more pronounced drop in classification accuracy.

Future Work
Future work could include applying the already-implemented Grad-CAM++ for more precise localization, testing robustness to additional perturbations (e.g., occlusion, blur, color changes), evaluating on more challenging datasets with higher variability, and using the show_grad_cam_errors_general function to analyze model focus in misclassified cases.

Ethics Statement

1. Introduction
Student names: Idan Avisar, Tomer Ben Aroush
Project title: Comparative Analysis of Neural Network Architectures for Yoga Pose Detection
Project description: This project compares multiple neural network architectures for detecting and classifying yoga poses. It evaluates the performance of pre-trained models such as ResNet and YOLO against a custom neural network, optimizes a selected architecture, and tests all models on multiple datasets to assess generalization and robustness.

2. Stakeholders
a) **Stakeholder groups:**

- Yoga practitioners, instructors, and studios - potential use for improving technique, enabling remote teaching, and managing classes.
- Software developers and AI researchers - models, results, and code can serve as a base for further development in pose estimation.
- Hardware and technology companies - potential integration of optimized models into devices like smartphones, cameras, or smart mirrors.

b) **Explanation to stakeholders:**
For the yoga community, the explanation would emphasize practical benefits, such as real-time feedback and remote learning. For developers and researchers, it would focus on methodologies, metrics, and comparative insights. For technology companies, it would highlight performance, efficiency, and ease of integration into products.

c) **Responsibility for explanation:**
The project team, led by the project manager or principal investigator, is responsible for delivering explanations. Communication may be delegated - technical details to developers/researchers, practical benefits to the yoga community, and commercial aspects to business representatives.

3. Ethical considerations:
Explanations must include disclaimers about the model's limitations and potential biases. For example, accuracy may vary across body types, skin tones, and lighting conditions. Tech companies must be informed about the training data sources, potential biases, and privacy implications. Ethical communication requires transparency about both the capabilities and limitations of the system to ensure responsible and safe use.

Github link : https://github.com/avisar1/ee0460217/tree/main

Youtube link: https://www.youtube.com/watch?v=Ck7_QeH6kl4

References

[1] yoga pose dataset: https://www.kaggle.com/code/aayushmishra1512/yoga-pose-detection/input

[2] R. R. Selvaraju, M. Cogswell, A. Das, et al., "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, pp. 618–626, 2017.

[3] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Improved visual explanations for deep convolutional networks," in *Proc. IEEE Winter Conf. Applications of Computer Vision (WACV)*, pp. 839–847, 2018**.**

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, et al., "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learning Representations (ICLR)*, 2021.